2011 International Conference on Document Analysis and Recognition

ICDAR 2011 Robust Reading Competition

Challenge 1: Reading Text in Born-Digital Images (Web and Email)

D. Karatzas, S. Robles Mestre, J. Mas, F. Nourbakhsh
Computer Vision Center
Universitat Autónoma de Barcelona
Barcelona, Spain
{dimos, srobles, jmas, farshad}@cvc.uab.es

P. Pratim Roy
Laboratoire d'Informatique
Université François Rabelais
Tours, France
partha.roy@univ-tours.fr

***Abstract*—This paper presents the results of the first Challenge of ICDAR 2011 Robust Reading Competition. Challenge 1 is focused on the extraction of text from born-digital images, specifically from images found in Web pages and emails. The challenge was organized in terms of three tasks that look at different stages of the process: text localization, text segmentation and word recognition. In this paper we present the results of the challenge for all three tasks, and make an open call for continuous participation outside the context of ICDAR 2011.**

Web images, email images, born-digital, text extraction, localization, segmentation, recognition

I. INTRODUCTION

Images are frequently used in Web pages and email messages to embed textual information. The use of images as text carriers stems from a number of needs, for example in order to beautify (e.g. titles, headings etc), to attract attention (e.g. advertisements), to hide information (e.g. images in spam emails used to avoid text-based filtering), even to tell a human apart from a computer (CAPTCHA tests).

The problem has been quantified in the past in terms of the use of image-text in Web pages [1, 2]. Past research has shown that a considerable amount of text on Web pages is presented in image form (17%), while an important fraction of this text (76%) is not to be found anywhere else in the Web page [1]. Taking into account that the very text that is presented in image form is more often than not semantically important (i.e. titles, headings, advertisements), one can get a feeling of the importance of the problem.

Automatically extracting text from born-digital images is an interesting prospect as it would provide the enabling technology for a number of applications such as improved indexing and retrieval of Web content, enhanced content accessibility, content filtering (e.g. advertisements or spam emails) etc.

Although there has been a lot of work on text extraction from complex images (video frames, real-scenes, book and magazine covers), little work has been published specifically focused on born-digital images [3, 4, 5]. While born-digital text images might seem similar to other complex text containers they present certain distinct characteristics. Born-digital images are inherently low-resolution (made to be transmitted online and displayed on a screen), they often suffer from compression artefacts and severe anti-aliasing while text is digitally created (over imposed) on the image. For the sake of comparison, real-scene images are high-resolution camera captured images that often present illumination problems and perspective transformations. Therefore, it is not necessarily true that methods developed for one domain would work in the other. The existence of a second challenge on real-scene images under this competition [6] serves exactly to quantify and qualify the differences between the real-scene and born-digital domains.

Of particular interest in the born-digital domain is the application of image-spam filtering, and there have been a few approaches that attempt to classify email images as legit or spam [7]. Generally such approaches are based on features such as the extent of text in the image (hence stop at rough text-localization) rather than attempting to extract and analyse the textual content [8]. Nevertheless, the benefits of using extracted textual content from spam images for filtering are substantial [9], even if limited recognition success is achievable.

In the rest of the paper we present the results over three independent tasks related to text extraction from born-digital images: text localization, text segmentation and word recognition. The structure of the competition is described in Section II. Section III describes the dataset and ground truth used for the competition, while Section IV details the performance evaluation methodology followed. In Section V we briefly present the participating methods, before results are presented in Section VI. We close this paper with offering some conclusions and suggestions in Section VII.

II. THE CHALLENGE

This Challenge was organised over three tasks: text localization, text segmentation and word recognition. The tasks selected reflect typical steps in the process of text extraction, but are not meant to be interpreted as a firm processing pipeline. On the contrary, independent participation in each of the tasks was encouraged.

The objective of Task 1: Text Localization, was to obtain a rough estimation of the text areas in the image, in terms of bounding boxes corresponding to parts of text (words or text lines). Task 2: Text Segmentation, aimed at the pixel level separation of text from the background. Finally, for Task 3: Word Recognition, we assumed that the locations (bounding

1520-5363/11 $26.00 © 2011 IEEE
DOI 10.1109/ICDAR.2011.295

boxes) of words in the image were known and sought the correct text transcriptions.

The Challenge was organized in open mode, meaning that we expected participants to run their own algorithms and provide us with the result files directly instead of executable programs. Participants committed not to modify their system during the test phase and we rest on the academic integrity of the participants when reporting final results.

The participating authors were required to register their interest by registering on the Challenge Web site. The training set and corresponding ground truth were made available on the 20th of March 2011 giving authors an ample period of more than two months to train their systems. The test set was made available to the registered authors in June 1st, and a period of three days was provided to submit results directly on the Challenge Web site. All submitted files were automatically validated at the time of uploading and confirmation of receipt was provided online to the users.

Since the Web site of the Challenge was opened we received 692 visits[1]. In total 31 users registered on the Web site and downloaded the datasets. Of these users we received 7 submissions[2] for Task 1 (Text Localisation), 3 submissions for Task 2 (Text Segmentation) and 1 submission for Task 3 (Word Recognition).

III. DATASETS

A new dataset was created for this Challenge. For creating the dataset we analysed 412 HTML documents, out of which 22 spam 75 ham (newsletters etc) emails. For the 315 Web pages analysed we selected a representative cross-cut of different page categories (News, Government, Commercial, Social, etc), replicating real-world statistics [10, 11]. All sites analysed were in English.

We extracted all the images that contained text, and selected 420 images (with size larger than 100x100 pixels), containing 3583 words of more than 3 characters for the training set and 102 images, containing 918 words, for the test set.

Ground truth information for each image was created manually all the way to the pixel level. The ground-truth specification is presented in [12] and captures information at different levels, from character parts up to text lines, and from pixel level labelling up to bounding boxes and transcriptions. The default ground truth format is XML as described in [12]. For simplicity we extracted individual aspects of the ground truth for the different tasks.

For Task 1 (Text Localization) we provided bounding boxes of words for each of the images. The ground truth was provided as separate text files (one per image for the 420 images) with each line specifying the coordinates and transcription of one word's bounding box. For Task 2 (Text Segmentation), the ground truth data was provided in the form of bi-level PNG images, with text pixels labeled 1 (white) and background pixels labeled 0 (black). For Task 3 (Word Recognition), we used all the words in our dataset with a length of 3 characters or more (3583 words in the training set). The words were cut from the original images, including a border of 4 pixels to maintain the immediate context (similarly to the MNIST character dataset), and were provided as a collection of image files, along with their corresponding ground-truth transcription.

The datasets will remain publicly available through the Web site of the Challenge, and will be submitted for archiving to the Web site of TC11.

IV. PERFORMANCE EVALUATION

A. Task 1 – Text Localisation

For evaluating the performance of Text Localization methods we implemented the methodology proposed by Wolf et al [13]. The aforementioned method is an improvement over simple bounding box area overlapping methods, in that it takes into account both the area precision and the precision at the level of detection counts, providing a way to create meaningful cumulative results over many images. In addition, it provides a way to deal with many-to-one and one-to-many relationships (e.g. many words in the ground truth, corresponding to a single detected text line in the results). We used the area recall and precision thresholds proposed in the original publication: $t_r = 0.8$, $t_p = 0.4$. For the one-to-many matches we used the proposed $f_{cs}(k)=0.8$ for $Match_G$, punishing the case of over-segmentation (many detected bounding boxes correspond to a single ground truth bounding box). For many-to-one matches we used a value of $f_{cs}(k)=1$ for $Match_D$, giving no punishment to the under-segmentation case (one detected bounding box corresponds to many ground truth bounding boxes). The rationale is that our ground truth is given at the level of words, and there are many cases where algorithms work by design at the text-line level. We did not want to penalise unevenly different behaviours.

B. Task 2 – Text Segmentation

For Text Segmentation evaluation, we implemented two metrics. One is the standard pixel level precision and recall metric, which we include only for completeness and backward compatibility but we do not use for the final ranking as it has known problems. The primary evaluation scheme we used is the one described in [12], as it is designed taking into account the final objective which is segmentation for recognition. The question of segmentation is not only how many pixels are misclassified but which ones. The main point is that it is not the same to dilate/erode a character by a few pixels and to add/remove these pixels to the character in a way that its shape changes. Standard precision and recall methods do not differentiate between the two cases, whereas the scheme used makes explicit such qualitative differences and can provide in depth analysis of the results.

C. Task 3 – Word Recognition

For the evaluation of Word Recognition results we implemented a standard edit distance metric, with equal costs for additions, deletions and substitutions. We calculate the

[1] Source: Google Analytics

[2] One of the authors failed to provide us a description of their method hence we only report results on 6 methods.

normalized edit distance between the ground truth transcription and the submitted transcription and sum over the dataset. To assist qualitative analysis, we also provide binary classification (recognised/not recognised) statistics.

V. PARTICIPATING METHODS

A brief description of all participating methods is provided here. Certain authors participated in more than one tasks, here we present the complete methodologies and we indicate which of the tasks were tackled by each submission.

A. "TH-TextLoc / TH-OCR", C. Yang, C. Liu and X. Ding

Department of Electronic Engineering, Tsinghua University Beijing, China

In this system, we adopt a three-step region-based method to localize text in web images. First, all source images are up-scaled using a bi-cubic interpolation algorithm. Connected components are then extracted from the image using adaptive local binarization. Neighbouring connected components are analysed and noise components filtered based on heuristics. The main idea for neighbouring component analysis is inspired by [14]. Then a discriminative SVM classifier is employed to classify the remaining text candidates based on individual connected component features including geometric features (aspect ratio, perimeter), shape features (compactness, contour roughness, number of holes) and stroke features (mean and variance of stroke widths).

Finally, we group text candidates into text regions using a projection histogram analysis. To refine these text regions, we employ a region shrinking and growing post-processing step. Text lines are separated into individual words using heuristic rules. The resulting text regions were submitted to Task 1 (Text Localisation).

As far as Task 3 (Word Recognition) is concerned, the main focus was on the text binarization rather than core OCR algorithms. All word images provided were normalized to the same height using a bi-cubic interpolation method; we set the height to 100. During the text binarization stage the text polarity is established based on a connected component analysis method. Then adaptive local binarization is used to create coarse text components followed by morphological opening to separate consecutive connected characters. Noise and bar-style components are filtered out in a post-processing step.

For recognition we used the OCR engine TH-OCR 2007 [15], which is an Asian multi-language recognition software developed mainly by our group from Tsinghua University in China. For OCR text segmentation is guided by the binary image while for recognition greyscale features are used. It should be noted that the recognition result is achieved without using any language model.

B. "TDM_IACAS", Y. Shao, C. Wang and Y. Zhang

Institute of Automation, Chinese Academy of Sciences, Beijing, China

The method is based on multi-scale connected component analysis. For the particular images of Challenge 1, three scales are considered, which are dependent on the size of the image. The scaling factors used are the following:

$$\begin{array}{cccc} s_1 = 0.5, & s_2 = 1, & s_3 = 2, & if\ width > 1600\ or\ height > 1600 \\ s_1 = 1, & s_2 = 2, & s_3 = 4, & else\ if\ width > 800\ or\ height > 800 \\ s_1 = 1, & s_2 = 3, & s_3 = 6, & otherwise \end{array}$$

In each of the scales, the And-Ridge and And-Valley images are calculated as detailed in [16]. Connected component analysis is then performed in the above images, and resulting components are subsequently classified as character and non-character. For the classification each component is first normalized to 24x24 pixels and then divided into 3x3 sub blocks. From each sub-block 8-direction gradient features are extracted resulting in a 72-dimensional feature vector. A set of binary SMV classifiers were trained for every character except {i, I, j, J, 1, l} as discriminating these characters from noise is difficult. These characters are recuperated in the following grouping step.

Each connected component labelled as character is considered as a seed component for a text string and similar components are sought on the left and right directions in an iterative fashion. Components are considered similar if (a) their height ratio is less than a threshold (set to 2.1 for this application), (b) the horizontal intersection distance normalised by the sum of heights is less than a threshold (set to 0.3). (c) The distance between the components, normalised by the minimum of the two heights is less than a threshold (set to 3) and (d) the RGB colour distance is smaller than a threshold (set to 60).

C. "OTCYMIST", D. Kumar and A.G. Ramakrishnan

Medical Intelligence and Language Engineering Laboratory, Indian Institute of Science, Bangalore, India

The R, G, B channels of the input image are separately binarised using Otsu's method, and connected components are extracted in each binary image and its complement one. An initial filtering of connected components is performed based on area, aspect ratio and Euler number. The remaining connected components of each plane and its complement are thinned and combined into a single plane.

Each component's centroid is used as a node for a graph, and a minimum spanning tree is calculated. Edges are removed if they are longer than 2.5 times the average edge length, to segment words. Isolated nodes are removed while further filtering within each word takes place to ensure height consistency of the components. A new minimum spanning tree is calculated with the remaining components and the process is repeated one more time.

The final components are merged in a single plane and provide the segmentation results. These are then grouped horizontally based on their centroid locations and height similarity. In a final post-processing step the resulting groups are split in separate text lines and words examining the vertical and horizontal distances between the participating components. The resulting bounding boxes provide the text localization results.

D. "SASA", C. Yi[1] and Y. Tian[2]

(1) Dept. of Computer Science, City Univ. of New York, USA
(2) Dept. of Electrical Engineering, City University of New York, USA.

The localization algorithm includes two steps, layout analysis and text classification. Layout analysis combines

two algorithms. First, the magnitude gradient difference (MGD) map [17] is calculated from the Laplacian map of the original image, and the image is horizontally partitioned into several sub-images, where multi-scale scanning is performed according to the heights of edge boundaries. This algorithm handles text with small font size. Second, adjacent character grouping [18] is used to find out all possible fragments of text strings, consisting of three or more neighbouring edge boundaries with approximately equal heights and in horizontal alignment. This algorithm handles text with large font size.

Layout analysis generates a set of candidate image patches which are subsequently classified as text and non-text. A cascade Adaboost classifier was learned from the training set including ground truth text and non-text patches. Gradients, edge density, and text stroke orientations are used to calculate feature maps and block patterns [19] are defined to calculate patch feature vectors. Patches classified as text are merged into localized text regions which provide the text localisation results.

Pixel level segmentation is performed based on the localized text regions. The external boundary of a text region is assumed to be background. For all pixels inside a text region, the mean color distances from pixels in the external boundary of the text region are calculated, obtaining a vector of mean color distances. The vector is sorted and the median value is taken as threshold. Each pixel is then classified as background if its mean color distance is smaller than the threshold and as foreground (text) otherwise.

E. "TextHunter", M. Shehzad Hanif[1,2], L. Prevost[2,3]

(1) Univ. of Engineering and Technology, Lahore, Pakistan
(2) Institut des Systèmes Intelligents et de Robotique (ISIR),CNRS, UPMC Université Paris, France (3) Laboratoire de Mathematiques Informatique et Applications (LAMIA),Université des Antilles et de la Guyane, France

The proposed approach to localize text in scenes and digital images is composed of two steps: text detection and fine localization of text regions. The text detector is a cascade of boosted classifiers trained using AdaBoost algorithm. In contrary to current research in object detection, we proposed to use heterogeneous (belonging to different families) weak classifiers in the boosting paradigm. The weak classifiers used in this work belong to: generative and discriminant, linear and nonlinear, parametric and non-parametric families of classifiers. To encode textual information, we have proposed two sets of features. One feature set is based on the contrast between foreground and background while the second feature set encodes shape and appearance of characters in a text region. These features are computed on an image (detection) window of small size which is further divided into 16 cells or blocks. Detection window size varies form 32x16 pixels to 288x144 pixels in 9 steps. Weak classifiers are trained in a feature space containing single and pair-wise features. The output of text detector is a set of detections at various scales along with the confidence level of the detector [20].

In the fine localization step, the detections in (quasi) horizontal direction are grouped and their confidence levels are added resulting in candidate text regions. Then, connected components are extracted by applying morphological operations on the canny edge map of each candidate text region. Next, connected components are validated using simple thresholds on features such as confidence level, edge density, height, width and aspect ratio. Later, validated connected components are grouped to form text lines and/or words. The thresholds on features used for validation of connected components and grouping are learnt on the training database using a genetic algorithm where the objective is to maximize the F-measure on the given training set. In the case of scene text (see Challenge 2 report [6]), features based on gradient information of connected components are also taken into consideration during validation.

F. "Textorter", S. Tehsin and A. Masood

Military College of Signals, National University of Science and Technology, Pakistan

In this method edges are first extracted from a greyscale image. Here existing edge detection techniques are tailored to extract low-contrast edges. Then morphological operators are applied on the image aiming to connect any broken edges. A filtering stage removes noise components based on their aspect ratio and size. Remaining components are classified as text or non text on the basis of features such as size, aspect ratio and binary transitions. Finally, bounding boxes are defined around the text area which provides the results for text localisation. For text segmentation, a local threshold is applied to the greyscale version of each of the resulting text regions to obtain a binary image. This method can deal with text of varying font, size and colour, but can only be used for horizontally aligned text.

VI. RESULTS AND DISCUSSION

A. Task 1 – Text Localization

The ranking metric used for the Text Localisation task is the Harmonic Mean calculated according to the methodology described in [13]. Precision and Recall are calculated cumulatively over the whole test set (all detections over all 102 images pooled together). Results for the Text Localisation task are shown in TABLE I.

In order to compute a baseline to compare the participating methods against we have used a commercial OCR software package to obtain text localization and word recognition (see further below) results. For this purpose we have used the ABBYY OCR SDK (version 10) [21]. Factory default parameters were used for pre-processing with the single exception of enabling the option for low resolution images since the resolution of born-digital images is typically below 100DPI. For text localization we have used the reported location of individual words.

There was a trend by most participating methods to under-segment text regions, producing text-line bounding boxes instead of word ones. The ground truth data we compared against are given at the level of word bounding boxes, but there is no particular segmentation level widely accepted as a "best choice" for text localization; hence either word or text-line level results should be considered as

correct. As explained in section IV, we adjusted the parameters of the evaluation method [13] in order to minimise any unjust penalisation of text-line bounding boxes.

TABLE I. TEXT LOCALIZATION RESULTS (%)

Method	Recall	Precision	Harmonic Mean
Textorter	69.62	85.83	76.88
TH-TextLoc	73.08	80.51	76.62
TDM_IACAS	69.16	84.64	76.12
OTCYMIST	75.91	64.05	69.48
SASA	65.62	67.82	66.70
Text Hunter	57.76	75.52	65.46
Baseline Method	*70.32*	*84.25*	*76.66*

It is observed that the commercial system we used as a baseline performed very well. A more detailed analysis of the data (see Figure 1) shows a key qualitative difference: the baseline system misses/rejects (Precision and Recall at the image level equal to zero) many more images than the submitted systems, but when it works, it works well. On the other hand, submitted systems intend to treat every single image, frequently returning wrong detections.

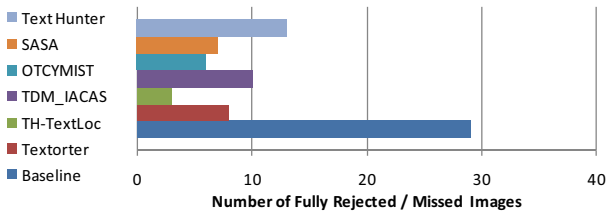

Figure 1. The baseline method rejects double the images than the rest.

B. *Task 2 – Text Segmentation*

In the case of text segmentation, we are evaluating results in two ways: our primary measure is based on the methodology detailed in [12], where quantitative evaluation is tightly connected with the quality of the result in terms of the integrity of the individual atoms on the page. Given a set of labeled connected components, this method reports the percentages of ground truth atoms that are "Well Segmented", "Merged" and "Broken" in the segmentation results. In the current scenario where the results are provided as binary images, the "Broken" category is not relevant and is not reported. The percentage of "Well Segmented" atoms is the primary metric for ranking the methods.

TABLE II. TEXT SEGMENTATION RESULTS (%)

Method	Well Segmented	Merged	Lost	Recall	Prec.	H. Mean
OTCYMIST	65.96	15.44	18.59	80.99	71.13	75.74
Textorter	58.73	32.53	8.73	65.20	62.50	63.82
SASA	42.71	10.70	46.57	71.93	54.78	62.19

In addition to the above measures, we also report a standard Precision and Recall metric at the pixel level, to maintain backwards compatibility to previously reported results. The ranking obtained using the harmonic mean of precision and recall agrees with the ranking obtained through our primary metric. Results are shown in TABLE II. and Figure 2. It can be appreciated that the main problem of Textorter (ranking second after OCTYMIST), is that of under-segmentation.

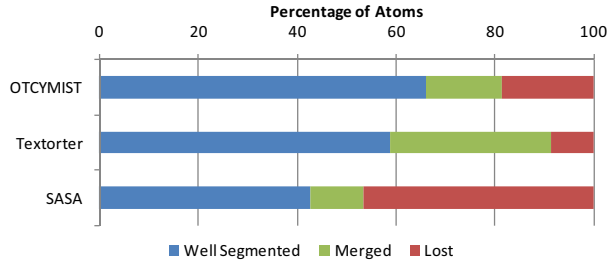

Figure 2. Atom-based Segmentation Results as per [12]

C. *Task 3 – Word Recognition*

In the case of Task 3, we only received a single participation. We report results here making a comparison to a baseline method as in the case of Task 1. For word recognition we have run the test images through the ABBYY OCR SDK (version 10) without any pre-processing and exported the result of the OCR in plain text. We report results here for completeness, but it should be stressed that the conclusions should be treated with caution as they are based on very limited data and a single participant.

The results of the Word Recognition task are shown in TABLE III. For each of the words we requested a single transcription, which was compared to the ground truth transcription based on a simple edit distance metric (equal weights for additions, deletions and substitutions). The edit distances calculated are normalized by the length of the ground-truth transcriptions and then summed over the 918 words of the test set. This is the primary metric we use for ranking.

TABLE III. WORD RECOGNITION RESULTS

Method	Total Edit Distance	Correctly Recognised Words (%)
TH-OCR	189.9	61.54
Baseline Method	*232.8*	*63.40*

In addition to the edit distance, we also report for completeness the percentage of words that were correctly recognized (no errors) by the two methods. Further analysis of the results showed that the majority of the correctly recognized words (48.8%) were common for both methods while 12.74% were recognized only by TH-OCR and 14.59% were recognized only by the baseline method.

Therefore, we can think of a notion of "objectively easy" and "objectively difficult" words in the case of born-digital images. This makes sense with hindsight as there are numerous (easy) images featuring single-colour text over single-colour background, and quite a few (difficult) images with photographic content and multi-colour text, but not so much in between.

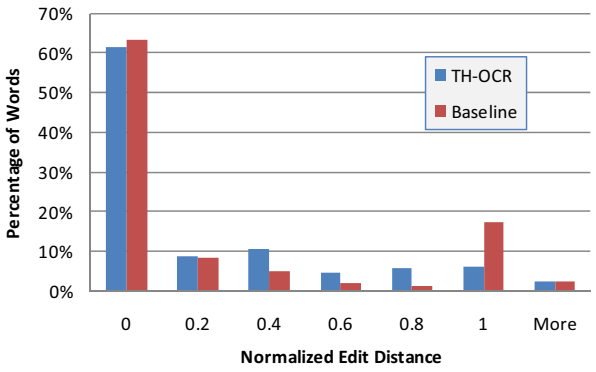

Figure 3. Histogram of normalized edit distances for Task 3.

Plotting the histogram of normalized edit distances (see Figure 3), we observe that errors are generally well distributed, with one notable peak of the baseline method at normalized edit distance equal to 1 (corresponding to all characters being changed, or equivalently to an empty response). Therefore the baseline method appears to have a good reject criterion, returning an empty string for cases it cannot recognize correctly.

VII. Concluding Remarks

The problem of text extraction from born-digital images is a very interesting one. Analysing these results we got a sense of two extremes when it comes to Web and email images. There is an easy to visualise category of "easy" images, usually featuring single colour text and background at a reasonable resolution for which text extraction works really well. At the same time, there are difficult images that present real challenges to state of the art methods. It would not be an exaggeration to say that when text extraction works, it works well; but when it fails, it fails in an effusive manner.

The results presented here are available online from the Web page of the competition in much more detail at the level of individual images for each task and method. We intend to maintain the Web site of the competition[3] open and we invite authors to register and submit results in a continuous mode. Performance evaluation is automatically provided with measures calculated as soon as new results are uploaded, making it easy to compare any new methods using the above standard evaluations. We plan to make available the tools for ground-truthing and dataset management in the near future, and extend the competition to other domains.

Acknowledgments

This work has been supported by the Spanish Ministry of Education and Science under projects TIN2008-04998, TIN2009-14633-C03-03, Consolider Ingenio 2010: MIPRCV (CSD200700018) and the grants 2009-SGR-1434 and RYC-2009-05031.

[3] http://www.cvc.uab.es/icdar2011competition/

References

[1] A. Antonacopoulos, D. Karatzas, J. Ortiz Lopez, "Accessing Textual Information Embedded in Internet Images", Proc. of SPIE, Internet Imaging II, San Jose, USA, Vol. 4311, pp. 198-205, January 2001

[2] T. Kanungo and C. Lee "What fraction of images on the web contain text ?", Int. Workshop on Web Document Analysis (WDA), pp 43-46, 2001

[3] D. Karatzas, A. Antonacopoulos, "Colour Text Segmentation in Web Images Based on Human Perception", Image and Vision Computing, Vol. 25, Issue 5, Elsevier, pp. 564-577, May 2007

[4] D. Lopresti, J. Zhou, "Locating and recognizing text in WWW images", Information Retrieval, 2, pp. 177–206, 2000

[5] S.J. Perantonis, B. Gatos and V. Maragos, "A novel Web image processing algorithm for text area identification that helps commercial OCR engines to improve their Web image recognition efficiency", Second Int. Workshop on Web Document Analysis (WDA2003), pp. 61-64, Edinburgh, Scotland, August 2003

[6] F. Shafait, A. Shahab, A. Dengel, "ICDAR 2011 Robust Reading Competition – Challenge 2: Reading Text in Scene Images", Proc. of 11th Int. Conf. on Document Analysis and Recognition, 2011

[7] N. Leavit, "Vendors Fight Spam's Sudden Rise", IEEE Computer, Vol. 40(3), pp 16-19, 2007

[8] H.B. Aradhye, G.K. Meyers, J.A. Herson, "Image analysis for efficient categorization of image-based spam e-mail", Proc. 8th Int. Conf. on Document Analysis and Recognition, pp 914-918, 2005

[9] G. Fumera, I. Pillai and F. Roli, "Spam fitlering based on the analysis of text information embedded into images", Proc. 1st Int. Symposium on Information and Communication Technologies, pp 291-296, 2003

[10] S. Robles Mestre, "A Study on the Information of Text Embedded in Web Images", Final Year Project Thesis, ETSE, UAB, 2009

[11] Online Source: http://www.internetworldstats.com/, May 2011

[12] A. Clavelli, D. Karatzas and J. Llados, "A Framework for the Assessment of Text Extraction Algorithms on Complex Colour Images", Proceedings of the 9th IARP Int. Workshop on Document Analysis Systems, ACM Press, pp. 19-28, Boston, MA, USA, 2010

[13] C. Wolf and J.M. Jolion, "Object Count / Area Graphs for the Evaluation of Object Detection and Segmentation Algorithms", Int. Journal of Document Analysis, vol. 8, no. 4, pp. 280-296, 2006

[14] B. Gatos, I. Pratikakis, K. Kepene, and S. Perantonis, "Text detection in indoor/outdoor scene images", Proc of CBDAR, pp. 127-132, 2005

[15] H. Liu, X. Ding," Handwritten Character Recognition Using Gradient Feature and Quadratic Classifier with Multiple Discrimination Schemes", Proc. of 8th Int. Conf. on Document Analysis And Recognition, pp.19-25, 2005

[16] Y. Shao and C. Wang, "Text Detection in Natural Images Based on Character Classification. Advances in Multimedia Information Processing", PCM 2010 Lecture Notes in Computer Science, 2011, Volume 6298/2011, 736-746

[17] T. Phan, P. Shivakumara and Chew Lim Tan, "A Laplacian Method for Video Text Detection," The 10th ICDAR, pp.66-70, 2009.

[18] C. Yi, and Y. Tian, "Text String Detection from Natural Scenes by Structure-based Partition and Grouping," IEEE Trans. on Image Processing, 2011

[19] X. Chen and A. L. Yuille, "Detecting and reading text in natural scenes," Proc. of IEEE Conf. on Computer Vision and Pattern Recognition, Vol. 2, pp. II-366 – II-373, 2004

[20] S. Muhammad Hanif, and L. Prevost, "Text Detection and Localization in Complex Scene Image using Constrained AdaBoost Algorithm", Proc. of 10th Int. Conf. on Document Analysis and Recognition, pp. 1-5, 2009

[21] Online Source: http://finereader.abbyy.com/, May 2011